Asymptotically Optimal Quantum Circuits for d-level Systems

Stephen S. Bullock[1]

stephen.bullock@nist.gov

Dianne P. O'Leary[1,3]

oleary@cs.umd.edu

Gavin K. Brennen[2]

gavin.brennen@nist.gov

October 15, 2004

[1] National Institute of Standards and Technology,
   Mathematical and Computational Sciences Division, Gaithersburg, Maryland 20899-8910

[2] National Institute of Standards and Technology, Atomic Physics Division, Gaithersburg, Maryland, 20899-8420

[3] University of Maryland, Department of Computer Science, College Park, Maryland 20742.

Abstract

As a qubit is a two-level quantum system whose state space is spanned by $|0\rangle$, $|1\rangle$, so a qudit is a d-level quantum system whose state space is spanned by $|0\rangle, \cdots, |d-1\rangle$. Quantum computation has stimulated much recent interest in algorithms factoring unitary evolutions of an n-qubit state space into component two-particle unitary evolutions. In the absence of symmetry, Shende, Markov and Bullock use Sard's theorem to prove that at least $C4^n$ two-qubit unitary evolutions are required, while Vartiainen, Möttönen, and Salomaa (VMS) use the QR matrix factorization and Gray codes in an optimal order construction involving two-particle evolutions. In this work, we note that Sard's theorem demands Cd^{2n} two-qudit unitary evolutions to construct a generic (symmetry-less) n-qudit evolution. However, the VMS result applied to virtual-qubits only recovers optimal order in the case that d is a power of two. We further construct a QR decomposition for d-multi-level quantum logics, proving a sharp asymptotic of $\Theta(d^{2n})$ two-qudit gates and thus closing the complexity question for all d-level systems (d finite.) Gray codes are not required, and the optimal $\Theta(d^{2n})$ asymptotic also applies to gate libraries where two-qubit interactions are restricted by a choice of certain architectures.

PACS: 03.67.Lx, 03.65.Fd   **AMS(MOS) subj:** 81P68, 65F25

Contents

1   Introduction

The dominant theoretical model of quantum computation is the quantum circuit [1] acting on quantum bits, or qubits [2]. A qubit is a two-level quantum system, whose complex Hilbert state space is spanned by kets $|0\rangle$ and $|1\rangle$. The labels within the ket are evocative of classical computer logic. Yet using qubits, these logical values may be placed in superposition, and moreover multiple qubits may be entangled. A *qudit* [3, 4] is a generalization to quantum computing of a classical multi-level logic [5]. We fix $d \geq 2$ throughout and consider the one-qudit state space

$$\mathcal{H}(1,d) \;=\; \mathbb{C}\,|0\rangle \oplus \mathbb{C}\,|1\rangle \oplus \cdots \oplus \mathbb{C}\,|d-1\rangle. \tag{1}$$

The decomposition is taken to be Hermitian orthonormal, and the n-qudit state space then becomes $\mathcal{H}(n,d) = \otimes_1^n \mathcal{H}(1,d) = \oplus_{c_1 c_2 \ldots c_n} \mathbb{C}\,|c_1 c_2 \ldots c_n\rangle$ with $c_1 c_2 \ldots c_n$ varying over all length n integers in base d.

For qudits and in particular qubits (i.e. $d = 2$,) computations are unitary evolutions of the state space. Of course, not all evolutions are computations. Rather, for an evolution to be unitary, the corresponding computation must be physically realizable with closed-system resources. Yet as most functions on bit-strings require exponentially many `AND-OR-NOT` gates, so too most unitary evolutions may only be realized with exponentially many quantum gates. The term computation is usually reserved for evolutions requiring few gates. In fact, a precise definition requires a family of evolutions indexed by n and an analysis of asymptotic complexities.

We say a function $h(n) \in \Omega[f(n)]$ if there is a constant C so that $h(n)$ is at least $Cf(n)$ for $n \geq 1$, and similarly $h(n) \in O[f(n)]$ if there is a second C so that $h(n)$ is at most $Cf(n)$. We say that $h(n) \in \Theta[f(n)]$ when both hold. Several choices of gate libraries are used in quantum circuit diagrams, but most admit asymptotically equivalent gate counts. We concentrate on two-qudit gates [6]. Thus, the complexity of a unitary evolution U is that number ℓ for which we have a minimum expression

$$U = U_{j_1 k_1}^1 U_{j_2 k_2}^2 \cdots U_{j_\ell k_\ell}^\ell \tag{2}$$

with each U_{jk}^p a two-qudit ($d^2 \times d^2$) operator acting exclusively on qudits j,k. An efficient computation should produce a family $\{U_n\}_{n=1}^\infty$ of unitary operators whose gate counts $\ell_{U_n} = h(n)$ satisfy $h(n) \in O(n^p)$, some $p > 0$. As an example of this formalism, put $\omega = \mathrm{e}^{2\pi i/2^n}$ and consider the n-qubit Fourier transform $\mathcal{F}_n = \frac{1}{\sqrt{2^n}} \sum_{j,k=0}^{2^n-1} \omega^{jk}\,|k\rangle\,\langle j|$.

Known circuits for $\mathcal{F}_n$ require $O(n^2)$ gates [7], so that the Fourier transform is an efficient quantum computation. The extension to qudits is likewise an efficient quantum computation [8].

It is typical to draw the factorization of U in Equation 1 as a quantum circuit, representing each qudit with a line and drawing a gate connecting qudits j, k for each U_{jk}^q. Physical implementation of symmetry-less evolutions is not practical when the number of qudits n is large, since the number of gates required scales exponentially in n. Yet circuits for generic unitaries are still of interest. First, they may improve subblocks of larger circuits through a process of peephole optimization: when many consecutive two-qudit gates act on a small collection of qudits, we compute the associated unitary evolution and substitute a circuit of the sort presented here in hopes of decreasing the total number of required operations. Second, they are useful in translating circuits from gate libraries that include three and multi-qudit gates to two-qudit gates when a physical system only conveniently allows for pairwise interactions. They may also be used to translate an arbitrary gate library into a fault-tolerant library of qudit gates [9]. Finally, we note that the symmetries that allow for polynomial-size quantum circuits are not well-understood. Thus, producing efficient symmetry-less circuits may provide insights into general design principles that might also be useful in constructing or optimizing computations.

For qubits, Shende, Markov and Bullock have shown that $\Omega(4^n)$ two-qubit gates are required, while a recent Letter [10] provided a $O(4^n)$ construction. Thus we have a sharp asymptotic for symmetry-less n-qubit unitary evolution: $\Theta(4^n)$ two-qubit gates are required. The result does *not* readily extend to qudits, even though qudit systems may be employed to emulate qubit systems and conversely. The lower bound generalizes to Cd^{2n} gates, but naïve emulations of the VMS circuit require asymptotically more gates than this. Indeed, the best prior constructive upper bound is $O(n^2 d^{2n})$ two-qudit gates [11].

The main result of our work is a constructive proof that $\Theta(d^{2n})$ two-qudit gates are required to implement an arbitrary n-qudit evolution without symmetry. En route, we also prove that $\Theta(d^n)$ two-qudit gates suffice for n-qudit state-synthesis. The algorithm that produces the quantum circuit is a variant of the *QR* matrix-decomposition, cf. [14, 15, 10]. Unlike an earlier qubit construction [10], it does not rely on a Gray code, either in base-two or base d.

The paper is organized as follows. First in §2, we review the justification of the lower bound of $\Omega(d^{2n})$ and then in §3 we discuss the inadequacy of qubit emulation of qudits. The remainder of the manuscript describes an algorithm for constructing a circuit involving two-qudit gates, carefully showing that the number of gates is $O(d^{2n})$. In §4, we review the idea of uniform control and extend it to the multi-level case. In particular, we show that a k-controlled one-qudit unitary may be implemented in $O(k)$ two-qudit gates, given sufficient ancilla (helper) qudits. In §5, we describe our state-synthesis algorithm and adapt it to a virtual Householder reflection $|\psi\rangle \mapsto \sqrt{\langle\psi|\psi\rangle}\,|j\rangle$ using singly controlled one-qudit operators. In §6, we present our qudit-native quantum circuit synthesis algorithm, and establish in §7 that it produces a universal circuit with at most $O(d^{2n})$ uniform controls.

2   The Lower Bound

The lower bound argument is similar to other lower bound arguments [12, 13] in quantum computing using Sard's theorem from smooth topology. The theorem (loosely) demands that almost no values of a smooth function are critical values. A well-known corollary (e.g. [13]) then demands that for a smooth map $f : M \to N$ that carries an m-dimensional manifold M into an n-dimensional manifold N for $m < n$, the set image(f) must be a measure zero subset of N.

We first set some notation. By default, upper case letters indicate either matrices or unitary operators. We use I_ℓ to denote an $\ell \times \ell$ identity matrix, and $A^\dagger = \overline{A}^T$ is the adjoint of A. Recall also the Lie theory notation $U(q) = \{U \in \mathbb{C}^{q \times q} ; UU^\dagger = I_q\}$. Suppose then that we consider an expression associated to a fixed circuit topology for two-qudit gates. Namely, suppose we factor a $U \in U(d^{2n})$ as in Equation 1. Suppose moreover that we take ℓ and the tuples (j_q, k_q) for $1 \le q \le \ell$ to be fixed. Then by varying the $U_{j_q k_q}^q$ in $U(d^2)$, we obtain a map of smooth manifolds $f : [U(d^2)]^\ell \to U(d^n)$. Now generically, $\dim[U(q)] = q^2$. Hence the smooth function implicit in the circuit diagram of Equation 1 carries a manifold of dimension ℓd^4 into a manifold of dimension d^{2n}. In order for the image to not be measure 0 for a fixed circuit diagram, we require $\ell \ge d^{2n-4}$. As there are only finitely many circuit topologies holding fewer than d^{2n-4} factors per Equation 1, we generically require $\Omega(d^{2n-4}) = \Omega(d^{2n})$ gates of the two-qudit library to realize symmetry-less unitary evolutions within $U(d^n)$.

A similar invocation of Sard's theorem produces a lower bound on circuit sizes for state synthesis. Here, the problem is to produce the most efficient possible circuit U capable of realizing generic $|\psi\rangle \in \mathcal{H}(n,d)$ from a fixed start-state typically chosen to be $|0\rangle$, i.e. building a small circuit for U so that $|\psi\rangle = U|0\rangle$. We claim that circuits for generic state-synthesis require $\Omega(d^n)$ gates. Indeed, $U|0\rangle$ is simply the first column of the matrix realization of U, and taking the column of a matrix is a smooth map. Hence we apply the Sard's theorem argument above to $\tilde{f} : U(d^2)^\ell \to \mathcal{H}(d,n)$, whence the result. Now a subcircuit of our universal unitary qudit-evolution circuit, described in Equation 6, is also capable of solving the state synthesis problem in $(d^n-1)/(d-1) \in O(d^n)$ two-qudit gates. Hence the qudit state-synthesis generically requires $\Theta(d^n)$ gates.

Theorem: *The following asymptotics hold for d multi-level quantum logic circuits. In each statement, d is fixed and the asymptotic is stated exclusively in terms of n.*

1. *Given a generic $|\psi\rangle$, constructing a quantum circuit for a unitary U such that $U|0\rangle = |\psi\rangle$ requires $\Theta(d^n)$ two-qudit gates.*

2. *Constructing a quantum circuit for a generic n-qudit unitary operator $U \in U(d^{2n})$ consisting of two-qudit gates requires $\Theta(d^{2n})$ two-qudit gates.*

As a remark, other gate libraries that are asymptotically equivalent to two-qudit gates might be better suited to certain problems. In reasonable cases, there should be a fixed upper bound on the number of library-gates required to realize a two-qudit unitary operator. For any such library a sharp asymptotic of $\Theta(d^{2n})$ gates is likewise required for generic unitary evolution. This holds in particular for $\{\text{local unitary}\} \sqcup \{\bigwedge_1(\sigma^x \oplus I_{d-2})\}$ [16].

3 Qubit Emulation is Insufficient

Consider two emulation schemes of qudits by qubits:

1. One might emulate each individual qudit with as few qubits as possible, so that the local qudit structure is respected.

2. One might rather pack the entire d^n dimensional n-qudit state into the smallest possible qubit state space, ignoring the local (tensor) structure.

We argue that the emulation circuit for Option 1 does not attain the lower bound asymptotic, while in essense Option 2 does not allow for circuit-level emulation at all.

In Option 1, label $\beta = \lceil \log_2 d \rceil$, so that β qubits are required to emulate a qudit. Now for the qubit circuit diagram, some multi-qubit gates will in fact be local to the qudit, while others are genuine two-qudit gates. Hence, if U is a $d^n \times d^n$ unitary matrix and the $O(2^{2\beta n})$ circuit is applied after splitting each qudit into β virtual qubits, we obtain an *upper* bound of $O(2^{2\beta n})$ two-qudit gates. Note that this asymptotic is worse than both $O(d^{2n})$ and even $O(n^k d^{2n})$ unless d is a power of two. (For $2^{2\beta} > d^2$ in this case, so the exponentials have distinct bases and are not asymptotically equivalent.) Thus, prior art does not suffice for the upper bound asymptotic.

We next consider Option 2. Note that n qudits may be viewed as d^n Hilbert space dimensions. Ignoring the local structure, a unitary evolution of $\mathcal{H}(n,d)$ may be realized as a subblock of a unitary evolution of $n\delta = \lceil n\log_2 d \rceil$ qubits rather than $n\beta = n\lceil \log_2 d \rceil$ as above. Indeed, with this form of emulation, it is true that $O(4^{n\delta}) = O(d^{2n})$ virtual two-qubit gates would suffice by earlier methods. However, in this mode of emulation a virtual two-qubit gate *need not* correspond to a two-qudit gate. Indeed, it might not even be a k-qudit gate for k small. Consider for example two-qutrit gate of the form $I_3 \otimes V$ acting on $\mathcal{H}(3,3)$, where $V \in U(3^2)$. This has a 9×9 block structure, but emulating such a unitary using qubits is more or less an arbitrarily difficult 5-qubit evolution. It is certainly not a two-qubit gate! Thus, although the mapping between Hilbert spaces is possible, tensor (Kronecker) product structures are not preserved.

There are several candidate systems for quantum computation where the physical subsystems encoding the quantum information have dimension $d > 2$. Examples include charge-position states in quantum dots [17], rotational and vibrational states of a molecule [18], truncated subspaces of harmonic oscillator states [19] and ground

4

electronic states of alkali atoms with total spin $F > 1/2$ [20]. Moreover, it is useful to allow d not a power of two. First, in many instances, the physics of the system can preclude encoding in a Hilbert space of arbitrary size. For example, in the case of encoding in alkali atoms the Hilbert space dimension of a single hyperfine manifold is $2F + 1$ so for bosonic atoms, d is never a power of two. [1] Second, there is evidence that the fault-tolerant threshold for quantum computation can be improved when using error correction codes on qudits with d prime [21].

4 Uniformly-Controlled One-Qudit Operators

Although the complexity bound is phrased in terms of two-qudit operators, our *QR* factorization algorithm in §6 produces a quantum circuit of k-controlled one-qudit operators. The majority are doubly or singly controlled. We next review how a k-controlled qudit operator may be realized in $O(k)$ two-qudit gates, given $r = \lceil (n-2)/(d-2) \rceil$ ancilla qudits.

Qudit Generalizations of CNOT

The most common two-qubit gate is the quantum controlled-not, due to its appearance in early papers on quantum computing and reversible classical computation. In case $d = 2$, this gate, denoted CNOT or $\bigwedge_1(\sigma^x)$, linearly extends the action of the classical CNOT on bit-strings within two-qubit kets. Thus CNOT applies a NOT (σ^x) iff the control bit carries one. So in two-qubits with control on the most significant, CNOT linearly extends $|00\rangle \mapsto |00\rangle$, $|01\rangle \mapsto |01\rangle$, $|10\rangle \mapsto |11\rangle$, and $|11\rangle \mapsto |10\rangle$. An extension to arbitrary d has been suggested [16].

We may view σ^x as addition mod 2, which generalizes as follows. If $c \in \mathbb{Z}/d\mathbb{Z}$ is a dit, then we use $(\oplus c)$ to (abusively) denote both the addition map $k \mapsto k \oplus c$ within $\mathbb{Z}/d\mathbb{Z}$ and also the one-qudit unitary operator given by the permutation matrix of this map. So for example in qutrits ($d = 3$,)

$$\oplus 1 = \text{INC} = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \tag{3}$$

A corresponding (unitary) permutation map INC is given by $\text{INC}|j\rangle = |(j+1)\bmod d\rangle$ for any base d.

Then the CINC (controlled-increment) gate applies INC iff the control qudit carries $d - 1$, i.e. in the case of most-significant qudit control

$$\text{CINC}|j,k\rangle = \begin{cases} |j,(k+1)\bmod d\rangle & j = d-1 \\ |j,k\rangle & j \neq d-1 \end{cases} \tag{4}$$

We take the $\oplus$ symbol in a circuit diagram to designate modular increment INC as in the $d = 2$ case, so that the usual symbol for CNOT in a d-level diagram now designates CINC.

Using the most recent argument [16], a second generalization of CNOT must be added to the qudit local unitary group $U(d)^{\otimes n}$ in order to recover exact universal qudit computation. For this, we label $\sigma^x \oplus I_{d-2}$ as that computation with $|0\rangle \leftrightarrow |1\rangle$ and all other $|j\rangle \mapsto |j\rangle$, $2 \leq j \leq d-1$. Then the appropriate second gate is $\bigwedge_1(\sigma^x \oplus I_{d-2})$, i.e. $\bigwedge([d-1T], \sigma^x \oplus I_{d-2})$ or $\bigwedge([Td-1], \sigma^x \oplus I_{d-2})$ in the expanded notation of the next section. Note that a CINC gate may be constructing using INC gates and $d-1$ copies of $\bigwedge_1(\sigma^x \oplus I_{d-2})$ [16]. Since a *QR* argument ibid. also produces any two-qudit operator with at most $O(d^2) = O(1)$ gates from the library {local unitary} $\sqcup \{\bigwedge_1(\sigma^x \oplus I_{d-2})\}$, the optimal asymptotics of the Theorem apply equally well to this library.

Uniformly-controlled one-qudit operations [23]

We next extend the idea of uniformly-controlled [23] V operations from qubits to qudits. Here, V is a one-qudit operator applied to a target qudit based on a string of $n - 1$ controls. Each control is either $*$, to denote a match with an arbitrary value (no control,) or is chosen to be one of $0, 1, \ldots, d-1$, to force a specific matching value (control.) Note that standard control and local operations may be used to emulate uniform control at low cost.

---

[1] The dimension of the total ground state Hilbert space including both manifolds corresponding to the two spin states of the valence electron may be a power of two e.g. ^{87}Rb and ^{133}Cs.
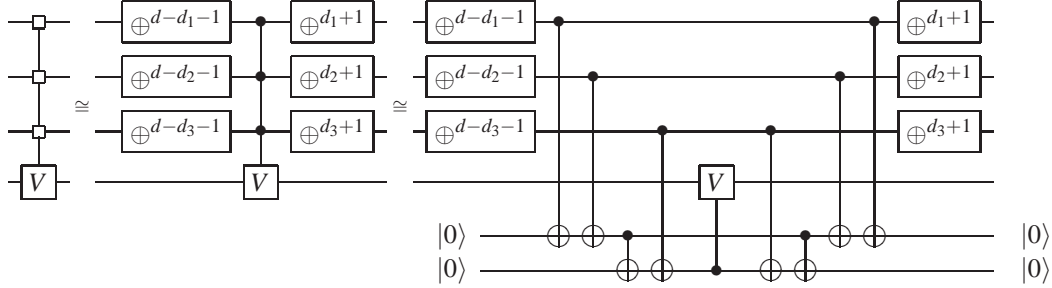
Figure 1: Qutrit ($d = 3$) emulation of $\bigwedge([d_1 d_2 d_3 T], V)$ using at most two-qudit operations. The general argument shows that $\bigwedge(C, V)$ requires at most $O(\#C)$ two-qudit gates. The rightmost circuit diagram is well-known [11, Figure 1]. Also, the controlled-V may be decomposed into local operators, CINCs, and $\bigwedge_1(\sigma^x \oplus I_{d-2})$ if the finer gate library is of interest [16].

In circuit diagrams, we will denote a uniform control triggering on an arbitrary $|j\rangle$ with a box, in contrast to the standard control denoted by a bullet that only triggers on state $|d-1\rangle$. One formal definition of uniform control is the following:

Definition 4.1 [Uniformly-controlled one-qudit operator $\bigwedge(C, V)$] Let V be a $d \times d$ unitary matrix, i.e. a one-qudit operator. Let $C = [C_1 C_2 \ldots C_n]$ be a length-n *control word* composed of letters from the alphabet $\{0, 1, \ldots, d-1\} \sqcup \{*\} \sqcup \{T\}$, with exactly one letter in the word being T. By $\#C$ we mean the number of letters in the word with numeric values (i.e., the number of controls,) and the set of control qudits is the corresponding subset of $\{1, 2, \ldots, n\}$ denoting the positions of numeric values in the word. We will say that a control word *matches* an n-dit string if each numeric value matches. Then the uniformly-controlled one-qudit operator $\bigwedge(C, V)$ is the n-qudit operator that applies V to the qudit specified by the position of T iff the control word matches the data state's n-dit string. More precisely, in the case when $C_n = T$, then

$$\bigwedge([C_1 C_2 \ldots C_{n-1} T], V) |c_1 c_2 \ldots c_n\rangle = \begin{cases} |c_1 \ldots c_{n-1}\rangle \otimes V |c_n\rangle, & c_j = C_j \text{ or } C_j = *, \ 1 \le k \le n-1 \\ |c_1 \ldots c_{n-1} c_n\rangle, & \text{else} \end{cases} \quad (5)$$

Alternatively, if $C_j = T$ ($j < n$,) we consider the unitary (permutation) operator χ_j^n that swaps qudits j and n. Thus, $\chi_j^n |d_1 d_2 \ldots d_n\rangle = |d_1 d_2 \ldots d_{j-1} d_n d_{j+1} \ldots d_{n-1} d_j\rangle$. We remark that $\chi_j^n = (\chi_j^n)^T = (\chi_j^n)^{-1}$. We apply the same permutation to $C = [C_1 C_2 \ldots C_{j-1} T C_{j+1} \ldots C_n]$, obtaining $\tilde{C} = [C_1 C_2 \ldots C_{j-1} C_n C_{j+1} \ldots C_{n-1} T]$ and we define $\bigwedge(C, V) = \chi_j^n \bigwedge(\tilde{C}, V) \chi_j^n$.

We note an earlier simulation [11] of $\bigwedge(C, V)$ in terms of two-qudit operations. In addition to n data qubits, we also require $r = \lceil (n-2)/(d-2) \rceil$ [11] ancilla qudits initially set to $|0\rangle$, as illustrated in Figure 1 for $\#C = 3$ and qutrits ($d = 3$.) The idea is to use local operations to interchange uniform with standard control. Then a sequence of CINC's appropriately targeting the r ancillas change the state of the last ancilla to $|d-1\rangle$ if and only if each control line carries $|d-1\rangle$. The entire operation follows by applying a singly-controlled V using this last ancilla and then mirroring the CINC pattern in order to disentangle the ancilla qudits from the data qudits.

5 Asymptotically Optimal Qudit State-Synthesis

The key component of our universal qudit circuit is a subcircuit interesting in its own right: given a state vector $|\psi\rangle \in \mathcal{H}(n, d) \cong \mathbb{C}^{d^n}$, we construct a sequence of p uniformly-controlled one-qubit operators depending on $|\psi\rangle$ such that

$$\prod_{k=1}^{p} \bigwedge[C(p-k+1), V(p-k+1)] |\psi\rangle = |0\rangle. \quad (6)$$

We remark that we use $V(p-k+1)$ instead of V_{p-k+1}, since the latter sort of subscript is often used to denote a target qudit while we intend the target to be labelled by the T symbol within the word $C(p-k+1)$.

6

n	♣-sequence, $d = 3$
1	♣
2	0♣, 1♣, 2♣, ♣♣
3	00♣, 01♣, 02♣, 0♣♣, 10♣, 11♣, 12♣, 1♣♣, 20♣, 21♣, 22♣, 2♣♣, ♣♣♣
4	000♣, 001♣, 002♣, 00♣♣, 010♣, 011♣, 012♣, 01♣♣, 020♣, 021♣, 022♣, 02♣♣, 0♣♣♣
	100♣, 101♣, 102♣, 10♣♣, 110♣, 111♣, 112♣, 11♣♣, 120♣, 121♣, 122♣, 12♣♣, 1♣♣♣
	200♣, 201♣, 202♣, 20♣♣, 210♣, 211♣, 212♣, 21♣♣, 220♣, 221♣, 222♣, 22♣♣, 2♣♣♣, ♣♣♣♣

Figure 2: Sample ♣-sequences for $d = 3$, i.e. qutrits.

Before continuing to construct this component of our universal circuit, we note that this subcircuit achieves optimal asymptotical qudit state-synthesis. The state-synthesis problem is to construct efficient (small) quantum circuits whose associated unitary U has $U|0\rangle = |\psi\rangle$ for some arbitrary but pre-determined $|\psi\rangle \in \mathcal{H}(n,d)$. For $d = 2$, several works address this topic, e.g. [24, 25, 26, 27]. For our subcircuit in Equation 6, note that

$$\prod_{k=1}^{p} \bigwedge [C(k), V(k)^\dagger] |0\rangle = |\psi\rangle \tag{7}$$

Our construction realizes any $|\psi\rangle$ in $p = (d^n - 1)/(d - 1) \in O(d^n)$ two-qudit gates, since also $\#C(k) \leq 1$ for all k. Given the $\Omega(d^n)$ lower bound of the introduction, we conclude that qudit state synthesis generically requires $\Theta(d^n)$ gates.

Finally, we briefly note our decision to index the sequence of $\bigwedge(C,V)$ so that the earlier indices appear on the right. There are two reasons for this. First, it means the index describes the operators in the order in which they are applied to $|\psi\rangle$, rather than the reverse. Second, the state-synthesis has received more attention than generalized Householder reductions in the literature, and note that the indices of Equation 7 do increase to the right.

One-qudit Householder Reflections

Earlier universal $d = 2$ circuits [14] relied on a *QR* factorization to write any unitary U as a product of *Givens rotations*, realized in the circuit as k-controlled unitaries [15]. Such Givens rotations V coincide with the identity matrix except in the pairwise intersection of rows j, k, with columns j, k. Here, the entries $V_{jj}, V_{jk}, V_{kj}, V_{kk}$ entries mimic those of a 2×2 unitary matrix. Thus, a Givens rotation is geometrically a rotation in the jk-plane. In the multi-level case, we use *Householder reflections* [22, §5.1] instead of Givens rotations, in order to take full advantage of the range of single qudit operators.

Thus, suppose $|\psi\rangle \in \mathcal{H}(1,d)$, perhaps not normalized, and suppose we wish to construct a unitary operator W such that $W|\psi\rangle$ is a multiple of $|0\rangle$. Standard formulas exist for constructing such W for real vectors. For a complex vector, these formulas become

$$\begin{cases} |\eta\rangle & = |\psi\rangle - \sqrt{\langle\psi|\psi\rangle} \dfrac{\langle 0|\psi\rangle}{|\langle 0|\psi\rangle|} |0\rangle \\ W & = I_d - (2/\langle\eta|\eta\rangle) |\eta\rangle\langle\eta| \end{cases} \tag{8}$$

Then indeed $W|\psi\rangle$ is a multiple of $|0\rangle$.

n-qudit State-Synthesis

We next describe the algorithm for realizing $\prod_{k=1}^{p} \bigwedge[C(p-k+1), V(p-k+1)] |\psi\rangle = |0\rangle$ with $\#C(k) \leq 1$ and $p = (d^n - 1)/(d - 1)$. The circuit topology has a recursive structure that we abstract into the following algorithm that generates the ♣-sequence ("club-sequence".)

Algorithm 1: $\{s_1, \ldots, s_p\} = $ **Make-♣-sequence**(d,n)

    % *We return a sequence of $p = (d^n-1)/(d-1)$ terms, with n letters each,*
    % *drawn from the alphabet* $\{0,1,\ldots,d-1,♣\}$.
    Let $\{\tilde{s}_j\}_{j=1}^{\tilde{p}} = $ **Make-♣-sequence** $(d,n-1.)$
    for $q = 0,1,\ldots,d-1$ **do**
       The next $(d^{n-1}-1)/(d-1)$ terms of the sequence are formed by prefixing the letter q to each
       term of the sequence $\{\tilde{s}_j\}$.
    end for
    The final term of the sequence is $♣^n$.

    Sample ♣-sequences that illustrate the construction are given in Figure 2. Note that the number of elements in the sequence equals the number of uncontrolled or singly-controlled one-qudit operators in the state-synthesis circuit. We choose to describe the circuit by iterating over the sequence. Thus, in order to produce the circuit, it suffices to describe how to extract the control word C from a term t of the ♣-sequence and how to determine V from the term and $|\psi_j\rangle$, where $|\psi_j\rangle = \prod_{k=1}^{j-1} \bigwedge[C(p-k+1),V(p-k+1)]\,|\psi\rangle$ is the partial product. This may be done as follows.

Algorithm 2: $\bigwedge(C,V) = $ **Single-♣Householder** (♣ term $t = t_1 t_2 \ldots t_n$, n-qudit state $|\psi_j\rangle$)

    Initialize $C = **\cdots*$
    % *Set the target:*
    Let ℓ be the index of the leftmost ♣ and set $C_\ell = T$.
    % *Set a single control if needed:*
    if t contains numeric values greater than 0,
       Let q be the index of the rightmost such value and set $C_q = t_q$.
    end if
    Given $|\psi_j\rangle = \sum_{k=0}^{d^n-1} \langle k|\psi_j\rangle\,|k\rangle$, form a one-qudit state $|\varphi\rangle = \sum_{k=0}^{d-1} \langle t_1 t_2 \ldots t_{\ell-1} k 00 \ldots 0|\psi_j\rangle\,|k\rangle$.
    Form V as one-qudit Householder such that $V|\varphi\rangle = |0\rangle$.

    Figure 3 illustrates the gate produced from the output C and V from the algorithm **Single-♣Householder**. Figure 4 illustrates the order in which these $\bigwedge(C,V)$ reflections are generated if we iterate over the ♣-sequence. Each node of the tree is labeled by a ♣-term and represents a Householder reflection defined by the three indicated elements of $|\psi\rangle$. After the reflection, the first element in the node remains and the others are zeroed. The reflections are applied by traversing the graph in depth-first order, left to right. To understand the controls, notice that the leftmost Householder on each layer of the graph requires no control. For example, the Householder 0♣♣ defined by elements $0j0$ ($j = 0,1,2$) is applied to 9 sets of elements: $0j1$ and $0j2$, all zeroed, and $1j0, 1j1, 1j2, 2j0, 2j1, 2j2$, as yet not zeroed. For the other Householder nodes, the control is indicated in boldface. For the leftmost Householder in a group of d siblings, we do not wish to touch elements in groups to the left of it. So we set the control to stay within the group. For example, the Householder labeled 10♣ is also applied to elements in 11♣ and 12♣. For other Householders in a group, the corresponding elements in groups to the left are completely zero, and in groups to the right are as yet unzeroed. Thus, for example, the Householder labeled 11♣ is applied to 01♣ (all zero since 0♣♣ has already been applied) and 21♣. We can formalize this argument to a proof of correctness as given in Appendix A.

Householder Circuits Retaining $|j\rangle \neq |0\rangle$

In the *QR* unitary-circuit application, we will need not only $|\psi\rangle \mapsto \sqrt{\langle\psi|\psi\rangle}\,|0\rangle$ but also $|\psi\rangle \mapsto \sqrt{\langle\psi|\psi\rangle}\,|j\rangle$ for any $j = d_1 d_2 \ldots d_n$. Rather than provide a new algorithm, we instead adapt our algorithm for a collapse onto $|0\rangle$ into an algorithm for collapse onto $|j\rangle$. The idea is to permute the elements to put j in position 0, apply **Single-♣Householder**, and then permute back. The rest of this subsection describes this in detail.

    We abusively continue to use $\oplus p$ for $1 \leq p \leq d-1$ to denote the one-qudit unitary operator that carries $|k\rangle \mapsto |(p+k)\bmod d\rangle$, i.e. $\oplus p = \texttt{INC}^p$. Given the d-ary expansion of j, we have $\otimes_{k=1}^n [\oplus d_k]\,|00\ldots0\rangle = |j\rangle$.
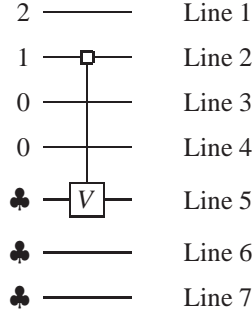
Figure 3: Producing a $\bigwedge(C,V)$ given V and a term of the $\clubsuit$-sequence, here $t = 2100\clubsuit\clubsuit\clubsuit$ for seven qudits. The algorithm for producing C places the V-target symbol T on the leftmost club, here line 5. The active uniform control must then be placed on the least significant line carrying a nonzero prior to line 5, here the 1 on line 2. (A control on lines 3 or 4 would not prevent the nonzero α_0 of $\left|\psi_j\right\rangle = \Sigma_{k=0}^{d^n-1}\alpha_k\left|k\right\rangle$ from creating new nonzero entries in previously zeroed positions.) Thus in this case, $C = *1**T**$. The V is chosen to zero all but one α_k for $k = 2100\ell00$.

Consider $\bigwedge(C,V)$, and define $\tilde{C}$ by

$$\tilde{C}_k = \begin{cases} *, & C_k = * \\ T, & C_k = T \\ (C_k + d_k)\bmod d, & C_k \in \{0,1,\ldots,d-1\} \end{cases} \tag{9}$$

Suppose also that $C_\ell = T$. Then noting that $(\oplus j)^\dagger = \oplus(d-j)$, we have the similarity relation

$$[\otimes_{k=1}^n \oplus (d_k)]\bigwedge(C,V)[\otimes_{k=1}^n \oplus (d-d_k)] \;=\; \bigwedge[\tilde{C},(\oplus d_\ell)V(\oplus d - d_\ell)] \tag{10}$$

This is the basis for the algorithm.

Algorithm 3: $\bigwedge(C,V) = \clubsuit$**Householder** $(\left|\psi\right\rangle, j, d, n)$

% *Reduce* $\left|\psi\right\rangle$ *onto* $\left|j\right\rangle$.
Let $j = d_1 d_2 \ldots d_n$.
Compute $\left|\varphi\right\rangle \;=\; [\otimes_{q=1}^n \oplus d - d_q] \left|\psi\right\rangle$.
Produce a sequence of uniformly-controlled one-qudit operators so that
   $\prod_{k=1}^p \bigwedge[C(p-k+1), V(p-k+1)]\left|\varphi\right\rangle = \left|00\ldots0\right\rangle$,
   using **Single-$\clubsuit$Householder** applied to each term of **Make-$\clubsuit$-sequence**(d,n).
Compute $[\otimes_{q=1}^n(\oplus d_q)]\bigwedge[C(p-k+1), V(p-k+1)][\otimes_{p=1}^n(\oplus d - d_q)] =$
   $\bigwedge[\tilde{C}(p-k+1), \tilde{V}(p-k+1)]$ using Equation 10

---

6 A Qudit-native QR-based Quantum Circuit Synthesis Algorithm

The asymptotically optimal qudit-universal circuit we present does not require Gray codes (Cf. [10].) Rather, it leans heavily on the optimal state-synthesis of §5. Since this state-synthesis circuit can likewise clear any length d^n vector using fewer than d^n single controls, the asymptotic is perhaps unsurprising. However, the recursive nature of our synthesis algorithm requires highly-controlled one-qudit unitary operators when clearing entries near the diagonal, and other highly-controlled one-qudit unitaries are needed to finish clearing each column. In presenting the algorithm, we highlight two themes:

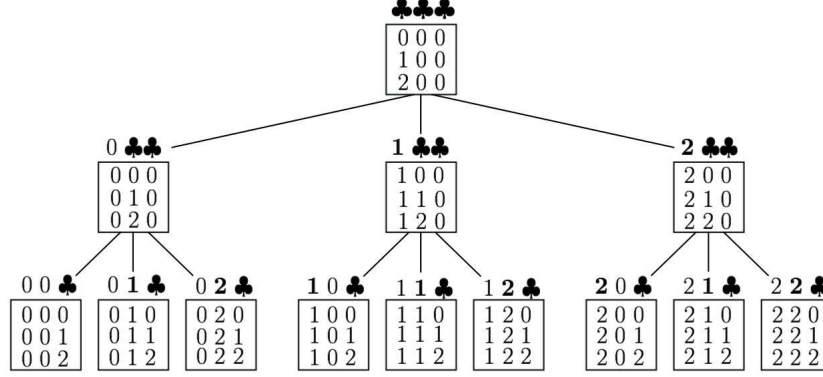- We process the size $d^n \times d^n$ unitary V in subblocks of size $d^{n-1} \times d^{n-1}$.

♣♣♣
```
0 0 0
1 0 0
2 0 0
```

0 ♣♣
```
0 0 0
0 1 0
0 2 0
```
1 ♣♣
```
1 0 0
1 1 0
1 2 0
```
2 ♣♣
```
2 0 0
2 1 0
2 2 0
```

0 0 ♣
```
0 0 0
0 0 1
0 0 2
```
0 1 ♣
```
0 1 0
0 1 1
0 1 2
```
0 2 ♣
```
0 2 0
0 2 1
0 2 2
```
1 0 ♣
```
1 0 0
1 0 1
1 0 2
```
1 1 ♣
```
1 1 0
1 1 1
1 1 2
```
1 2 ♣
```
1 2 0
1 2 1
1 2 2
```
2 0 ♣
```
2 0 0
2 0 1
2 0 2
```
2 1 ♣
```
2 1 0
2 1 1
2 1 2
```
2 2 ♣
```
2 2 0
2 2 1
2 2 2
```

Figure 4: Using the ♣-sequence for $d = 3$, $n = 3$ to generate Householder refections to reduce $|\psi\rangle$ to a multiple of $|0\rangle$. Each node is labeled by a ♣-term and represents a Householder reflection $\bigwedge(C, V)$. The control is indicated by the boldface entry in the label. The $\bigwedge(C, V)$ is constructed to zero the components of the last two indices in each node, using the component of the top entry.

- Due to rank considerations, at least one block in each block-column of size $d^n \times d^{n-1}$ must remain full rank throughout.

Hence, we cannot carelessly zero subcolumns. One solution is to triangularize the $d^{n-1} \times d^{n-1}$ matrices on the block diagonal, recursively. We also note that only $O(n^2 d^n)$ fully $(n-1)$ controlled one-qudit operations appear in the algorithm, which is allowed when working towards an asymptotic of $O(d^{2n})$ controls total.

The organization for the algorithm is then as follows. Processing (triangularization) of V moves along block-columns of size $d^n \times d^{n-1}$ from left to right. In each block-column, we first triangularize the block $d^{n-1} \times d^{n-1}$ block-diagonal element, perhaps adding a control on the most significant qudit to a circuit produced by recursive triangularization. After this recursion, we zero the blocks below the block-diagonal element one column at a time. For each column j, $0 \leq j \leq d^{n-1} - 1$, the zeroing process is to collapse the $d^{n-1} \times 1$ subcolumns onto their j^{th} entries, again adding a control on the most significant qudit to prevent destroying earlier work. These subcolumn collapses produce the bulk of the zeroes and are done using ♣**Householder**. After this, fewer than d entries remain to be zeroed in the column below the diagonal. These are eliminated using a uniformly-controlled reflection containing $n - 1$ controls and targeting the top line. With the appropriate one-qudit Householder, the diagonal entry will zero lower nonzero terms while the older zeroes in the column are protected by the controls on the lower $n - 1$ lines.

We now give a formal statement of the algorithm. We emphasize the addition of controls when previously generated circuits are incorporated into the universal circuit (i.e. recursively telescoping control.)

Algorithm 4: Triangle(U, d, n)

if $n = 1$ **then**
    Triangularize U using a *QR* reduction.
else
    Reduce top-left $d^{n-1} \times d^{n-1}$ subblock using **Triangle**$(*, d, n-1)$, (writing output to bottom $n-1$ circuit lines)
    for $k = 0, 1, \ldots, d-1$ **do**   % *Block-column iteration*
      for columns $j = kd^{n-1}, \ldots, [(k+1)d^{n-1} - 1]$ **do**
        for $\ell = (k+1), \ldots, (d-1)$ **do** % *Block-row iterate*
          Use ♣**Householder** to zero the column entries $(k+\ell)d^{n-1}, \ldots, [(k+\ell+1)d^{n-1} - 1]$,
            leaving a nonzero entry at $(k+\ell)c_2 \ldots c_n$ for $j = c_1 c_2 \ldots c_n$ and
            adding $|k+\ell\rangle$-uniform control on the most significant qudit.
        end for
        Clear the remaining nonzero entries below diagonal using one $\bigwedge[Tc_2 \ldots c_n, V]$.
      end for   % *All subdiagonal entries zero in block-col*
      Use **Triangle**$(*, d, n-1)$ on the $d^{n-1} \times d^{n-1}$ matrix at the $(k+1)^{\text{st}}$ block diagonal
        adding $|k+1\rangle$-uniform control to the most significant qudit.
    end for
end if-else

To generate a circuit for a unitary operator U, we use **Triangle** to reduce U to a diagonal operator $W = \sum_{j=0}^{d^n-1} e^{i\theta_j} |j\rangle \langle j|$. Now V and $U = WV$ would be indistinguishable if a von Neumann measurement $\{|j\rangle \langle j|\}_{j=0}^{d^n-1}$ were made after each computation. However, the diagonal is important if U is a computation corresponding to a subblock of the circuit of a larger computation with other trailing, entangling interactions. In this case, Figure 1 makes clear how to build a circuit for a controlled relative phase $I_{d^n} + (e^{i\theta} - 1)|j\rangle \langle j|$ in $O(n)$ gates. Since W has only $O(d^n)$ such phases, the corresponding circuit for W costs $O(nd^n)$ two-qudit gates and as such is asymptotically irrelevant to $\Theta(d^{2n})$.

7 Counting Gates and Controls

Let $h(n, k)$ be the number of k-controls required in the **Single-♣Householder** reduction of some $|\psi\rangle \in \mathcal{H}_n$. Then clearly $h(n, k) = 0$ for $k \geq 2$. Moreover, each 0-control results from an element of the ♣-sequence of the form $00 \ldots 0$♣♣$\ldots$♣, and there are n such sequences. Thus, since the number of elements of the ♣-sequence is $(d^n - 1)/(d-1)$, we see that

$$
\begin{cases}
h(n, 1) &= & (d^n - 1)/(d-1) - n \\
h(n, 0) &= & n
\end{cases}
\tag{11}
$$

We next count controls in the matrix algorithm **Triangle** of §6. We break the count into two pieces: g for the work outside the main diagonal blocks and f for the work within.

Let $g(n, k)$ be the number of controls applied in operations in each column that zero the matrix below the block diagonal; this is the total work in the **for** j loops of **Triangle**. We use **Single-♣Householder** $d(d-1)d^{n-1}/2$ times since there are $d(d-1)/2$ blocks of size $d^{n-1} \times d^{n-1}$ below the block diagonal, and we add a single control to those counted in h. The last statement in the loop is executed $d^n - d^{n-1}$ times. Therefore, letting δ_j^k be the Kronecker delta, the counts are

$$
g(n, k) \;=\; \delta_k^{n-1}(d^n - d^{n-1}) + \frac{1}{2}d(d-1)d^{n-1}h(n-1, k-1)
\tag{12}
$$

Supposing $n \geq 3$, then we see that

$$
g(n, k) \;=\; \begin{cases}
d^n - d^{n-1}, & k = n-1 \\
0, & n-1 \leq k \leq 3 \\
\frac{1}{2}d^n(d^{n-1} - 1) - \frac{1}{2}d^n(d-1)(n-1), & k = 2 \\
\frac{1}{2}d^n(d-1)(n-1), & k = 1 \\
0, & k = 0
\end{cases}
\tag{13}
$$

11

d $\backslash$ n	2	3	4	5	6	7	8	9	10
2	5	17	39	74	125	195	287	404	549
3	40	285	1 140	3 370	8 820	17 535	33 880	60 660	102 240
4	220	3 240	22 176	100 000	345 060	987 840	2 464 000	5 528 736	11 407 500
5	1 040	32 130	379 776	2 631 500	12 931 920	49 999 110	161 960 960	457 946 136	
6	4 560	301 239	6 220 032	66 768 750	470 221 200				
7	19 200	2 757 807	100 279 728	1 676 043 750					
8	79 040	24 994 494	1 608 794 112						
9	321 280	225 584 676							
10	1 296 640	2 032 525 629							
11	5 212 160	1 120 813 409							
12	20 904 960								

Figure 5: Total number of uniform control boxes in the new universal circuit as a function of the level d and number of qudits n

Finally, let $f(n,k)$ be the total number of k-controlled operations in the **Triangle** reduction, including the block diagonals. This work includes that counted in g, plus a recursive call to **Triangle** before the **for** k loop, plus $(d-1)$ calls within the k loop, for a total of

$$f(n,k) \;=\; g(n,k) + f(n-1,k) + (d-1)f(n-1,k-1), \tag{14}$$

with $f(n,0) = 1$ and $f(1,k) = 0$ for $n,k > 0$.

Using the recursive relation of Equation 14 and the counts of Equation 13, we next argue that **Triangle** has no more than $O(d^{2n})$ controls. Two lemmas are helpful.

Lemma 7.1 *For sufficiently large n, we have $f(n,k) \le d^{2n-k+4}$.*

Proof: By inspection of Equation 13, we see that $g(n,k) \le (1/2)d^{2n-k+2}$ for all k and n large. Now $f(n,0) = 1$, which we take as an inductive hypothesis while supposing $f(n-1,\ell) \le d^{2n-2-\ell+4} = d^{2n-\ell+2}$. Thus, using the recursion relation of Equation 14,

$$
\begin{aligned}
f(n,k) \;\le\; & \tfrac{1}{2}d^{2n-k+2} + d^{2n-k+2} + (d-1)d^{2n-k+3} \\
=\; & d^{2n-k+4}\left(\tfrac{1}{2d^2} + \tfrac{1}{d^2} + 1 - \tfrac{1}{d} \right)
\end{aligned}
\tag{15}
$$

Now since $d > 3/2$, we must have $\frac{1}{d} > \frac{3}{2d^2}$, whence an inductive proof of the result. $\square$

Lemma 7.2 $\sum_{k=0}^{n-1} kf(n,k) \in O(d^{2n})$.

The proof of Lemma 7.2 follows from checking $\sum_{k=0}^{n-1} kd^{2n-k} \in O(d^{2n})$. The latter fact follows from either explicitly computing the sum by deriving the appropriate geometric series or alternately using integral comparison. Thus, the total number of control boxes in the circuit digram grows as $O(d^{2n})$. The theorem of the introduction asserting a size $O(d^{2n})$ universal circuit composed of two-qudit gates follows, given the commentary of §4 on decomposing a k-controlled one-qudit operator into two-qudit gates.

Figure 5 shows actual counts of uniform control boxes for specific instances of d, n. These are illuminating given that Lemma 7.1 overestimates the number of k-controls for most k. These counts are obtained using a C++ implementation of the recursion presented in this section and have been verified by an explicit MatLab implementation of the entire circuit synthesis algorithm for small d, n.

8   Conclusions

We conclude with some remarks. Locality in quantum mechanics is a function of the tensor (Kronecker) product structure of the state space in question. In quantum computing, the Hilbert space factors are often finite dimensional. Measuring difficulty by counting two-particle interactions, we have generalized a recent optimal asymptotic of $\Theta(2^{2n})$ for two-level quantum bits to a new optimal asymptotic $\Theta(d^{2n})$ for d-level quantum dits.

The result is exponentially better (asymptotically) than that obtained by emulating such qudits with qubits, given $d \neq 2^{\ell}$. This arises since the tensor decompositions are incompatible, except in the case that d is a power of 2.

Multi-level quantum logics have been proposed as an alternative to qubits due to the trade-off in the tensor structure. For $d > 2$, there is a larger space of local operations, and fewer entangling gates might be required to realize a target quantum computation (unitary evolution [8].) This work has moreover demonstrated that such a benefit does not scale with the number of particles n, but rather must consist (at most) of a constant factor reduction in the number of required entangling gates. However, our result only applies to symmetry-less evolutions, and particular computations might be better suited to certain multi-level and tensor structures on Hilbert space than others.

Implementations: `MatLab` source code ("`.m files`") have been included in the posting of this docoument to `http://www.arxiv.org`. Please download the source format and consult the `README` file. There is also a short `C++` program implementing the recursive control box counts. These files are very useful in understanding §6.

NIST disclaimer. Certain commercial equipment or instruments may be identified in this paper to specify experimental procedures. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology.

References

[1] D. Deutsch, Quantum Computational Networks, *Proc. R. Soc. London A* **425**, 73 (1989).

[2] R. P. Feynman, The Computer as a Physical System: a microscopic quantum-mechanical Hamiltonian model of computers as represented by Turing-machines, *Found. Phys.* **16**, 507 (1986),
P. Benioff, *J. Stat. Phys.* **22**, 563 (1980).

[3] D. Gottesman, Fault-Tolerant Quantum Computation with Higher-Dimensional Systems, *Chaos, Solitons Fractals* **10**, 1749 (1999). `http://www.arxiv.org/abs/quant-ph/9802007`

[4] R. Blume-Kahout, C.M. Caves, I.H. Deutsch, Climbing Mount Scalable: Physical Resource Requirements for a Scalable Quantum Computer, *Found. Phys.* **32**, 1641 (2002).

[5] E.L. Lawler, Jour. ACM **11**, 431 (1964).
M. Fujita, Y. Matsunaga, and M. Cieseilski, Mulit-Level Logic Optimization, Chap. 2 of *Logic Synthesis and Verification,* edited by Soha Hassoun and Tsutomu Sasao, Kluwer Academic Press, Norwell Massachusettes, 2001.

[6] D.P. DiVincenzo, Two-qubit Gates are Universal for Quantum Computation, *Phys. Rev. A* **51**, 1015 (1995).

[7] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*, (Cambridge Univ. Press, 2000.)

[8] P. Hoyer, Efficient Quantum Transforms, `http://www.arxiv.org/abs/quant-ph/9702028`

[9] E. Knill, Non-binary Unitary Error Bases and Quantum Codes, `http://www.arxiv.org/abs/quant-ph/9608048`

[10] J.J.Vartiainen, M.Möttönen, M.M.Salomaa, Efficient Decomposition of Quantum Gates *Phys. Rev. Lett.* **92**, 17902 (2004).

[11] A. Muthukrishnan and C.R.Stroud Jr., Multivalued Logic Gates for Quantum Computation *Phys. Rev. A* **62**, 052309 (2000).

[12] V.V. Shende, I.L. Markov, S.S. Bullock, Minimal Universal Two-qubit Controlled-not Based Circuits, *Phys. Rev. A* **69** 062321 (2004). `quant-ph/0308033`

[13] S.S.Bullock and I.L. Markov, Asymptotically Optimal Circuits for Arbitrary n-qubit Computations, *Quant. Inf. and Comp.* **4** 27 (2004).

[14] A. Barenco, C. Bennett, R. Cleve, D. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. Smolin, and H. Weinfurter, Elementary Gates for Quantum Computation, *Phys. Rev A* **52** 3457 (1995).

[15] G. Cybenko, Reducing Quantum Computations to Elementary Unitary Operations, *Comp. in Sci. and Eng.* **27** March/April 2001.

[16] G.K.Brennen, D.P.O'Leary, S.S.Bullock, Criteria for Exact Qudit Universality, `http://www.arxiv.org/abs/quant-ph/0407223`

[17] S.G. Schirmer, A.D. Greentree, and D.K.L.Oi, Implementation of Controlled Multi-qudit Operations for a Solid-state Quantum Computer Based on Charge Qudits, `http://www.arxiv.org/abs/quant-ph/0305052`.

[18] E. A. Shapiro, I. Khavkine, M. Spanner, and M. Yu. Ivanov, Strong-field Molecular Alignment for Quantum Logic and Quantum Control, *Phys. Rev. A* **67** 013406 (2003).

[19] S.D. Bartlett, H. de Guise, and B.C. Sanders, Quantum Encodings in Spin Systems and Harmonic Oscillators, *Phys. Rev. A* **65** 052316 (2002).

[20] G. Klose, G. Smith, and P. S. Jessen, Measuring the Quantum State of a Large Angular Momentum, *Phys. Rev. Lett.* **86** 4721 (2001).

[21] D. Aharonov, Presented at *Conference on Quantum Information: Entanglement, Decoherence and Chaos*, Institute for Theoretical Physics, Santa Barbara, 2001 (unpublished).

[22] G.H. Golub and C. van Loan, *Matrix Computations*, Johns Hopkins Press, 1989.

[23] M. Möttönen, J. J. Vartiainen, V. Bergholm, M. M. Salomaa, Quantum circuits for general multiqubit gates, *Phys. Rev. Lett.* **93** 130502 (2004). `http://www.arxiv.org/abs/quant-ph/0404089`.

[24] M. Möttönen, J. J. Vartiainen, V. Bergholm, M. M. Salomaa, Transformation of quantum states using uniformly controlled rotations, `http://www.arxiv.org/abs/quant-ph/0407010`

[25] V.V. Shende, S.S. Bullock, I.L. Markov, A Practical Top-down Approach to Quantum Circuit Synthesis, `http://www.arxiv.org/abs/quant-ph/0406176`

[26] E. Knill, Approximation by Quantum Circuits, `http://www.arxiv.org/abs/quant-ph/9508006`

[27] D. Deutsch, A. Barenco, A. Ekert, Universality in Quantum Computation, *Proc. R. Soc. London A* **449**, 669 (1995).

A   Proof of Correctness for State-Synthesis

We sketch the proof of correctness of the Algorithm for state-synthesis employed to attain Equation 6:

$$\prod_{k=1}^{p} \bigwedge [C(p-k+1), V(p-k+1)] \, |\psi\rangle \; = \; |0\rangle \tag{16}$$

Given the Algorithm, $p = (d^n - 1)/(d - 1)$ is the number of elements of the ♣-sequence. Suppose for clarity that $|\psi\rangle$ is generic, so that no amplitudes (components) are zero at the outset. Then it would suffice to prove (i)

that each operator $\bigwedge[C(j), V(j)]$ introduces $d-1$ new zeroes into the state $|\psi_{j+1}\rangle$ not present in $|\psi_j\rangle$ and (ii) $\bigwedge[C(j), V(j)]$ does not act on previously zeroed entries. The assertion (i) is straightforward and left to the reader. However, the second assertion is *false*. Rather, the uniformly-controlled one-qudit operators do act on previously zeroed entries, but they act in such a way that only linear combinations of these zeroes are ever introduced as new amplitudes. The discussion below makes this assertion precise and proves it.

To facilitate this, we label $S = \{0, 1, \ldots, d^n - 1\}$ the index set and introduce the notation $S_*(j)$ for the set of component indices of $|\psi_j\rangle$ that have not explicitly been reduced to a zero by some $\bigwedge[C(k), V(k)]$, $k \leq j$. We label $S[C(j)]$ to be the set of control indices of $C(j)$, per Definition 4.1. Also, define ℓ by $C(j)_\ell = T$. Now there is a group action of $\mathbb{Z}/d\mathbb{Z}$ on the index set S corresponding to addition mod d on the ℓth dit:

$$c \bullet_\ell c_1 c_2 \ldots c_n = c_1 c_2 \ldots c_{\ell-1}(c_\ell + c \text{ mod d})c_{\ell+1} \ldots c_n \tag{17}$$

Since the operator $V(j)$ is applied to qudit ℓ, the amplitudes (components) of $|\psi_{j+1}\rangle$ are either equal to the corresponding amplitude of $|\psi_j\rangle$ or else are linear combinations of the $|\psi_j\rangle$-amplitudes whose indices lie in the $\mathbb{Z}/d\mathbb{Z}$ orbit contained in $S[C(j)]$. Formally, we have proven the following Proposition.

Proposition A.1 *Suppose*

$$(\mathbb{Z}/d\mathbb{Z}) \bullet_\ell S_*(j) \cap S[C(j)] \subseteq S_*(j) \cap S[C(j)] \tag{18}$$

(We remark that should the inclusion hold, then it is an equality.) Then $|\psi_{j+1}\rangle$ has at least $d-1$ more zero amplitudes than $|\psi_j\rangle$.

The final question is how one proves the appropriate set inclusions. The point is to carefully understand the structure of $S_*(j)$. We will eventually prove that $S_*(j)$ is the union of the three sets $R_1(j)$, $R_2(j)$, and $R_3(j)$ below. However, we define them independently, as the induction technically requires the decomposition at the jth step to avoid mixing as the next operator is applied.

Definition A.2   Suppose the jth term of the ♣-sequence is given by $c_1 c_2 \ldots c_{\ell-1}♣\ldots♣$. We have $C(j)$ the corresponding control word, with $C(j)_\ell = T$. Consider the following three sets, noting $R_1(j)$ may be vacuous.

$$
\begin{aligned}
R_1(j) &= \bigsqcup_{q=0}^{\ell-2} \left\{ c_1 c_2 \ldots c_q k 00 \cdots 0 \; ; \; k < c_{q+1}, k \in \{0, 1, \ldots, d-1\} \right\} \\
R_2(j) &= \left\{ c_1 \cdots c_{\ell-1} k 00 \ldots 0 \; ; \; k \in \{0, 1, \ldots, d-1\} \right\} \\
R_3(j) &= \left\{ f_1 \cdots f_{\ell-1} k_\ell k_{\ell+1} \ldots k_n \; ; \; f_1 f_2 \ldots f_{\ell-1} > c_1 c_2 \cdots c_{\ell-1}, k_* \in \{0, 1, \ldots, d-1\} \right\}
\end{aligned} \tag{19}
$$

Remark A.3   These sets may be interpreted in terms of Figure 4. Recall the figure recovers the ♣-sequence by doing a depth-first search of an appropriate tree. In this context, $S_*(j)$ is the set of nonzero components of $|\psi_j\rangle$ at the jth node. The subset $R_3(j)$ results from indices that lie in nodes not yet traversed, loosely below the present node in the tree or to the right. The set $R_2(j)$ is precisely the set of indices in the current node, node j. The set $R_1(j)$ is the set of indices of elements that have been previously used to zero other elements and still remain nonzero themselves; it is the set of indices of elements that were always at the top of nodes already traversed in the depth-first search. Thus, $R_1(j)$ is loosely a set of entries within nodes to the left and perhaps above node j. $\diamond$

Lemma A.4 *Let $C(j)$, ℓ, be as above, and label $\tilde{S}_*(j) = R_1(j) \sqcup R_2(j) \sqcup R_3(j)$. Then*

$$(\mathbb{Z}/d\mathbb{Z}) \bullet_\ell \tilde{S}_*(j) \cap S[C(j)] \subseteq \tilde{S}_*(j) \cap S[C(j)] \tag{20}$$

Proof: Due to the choice of a single uniform control on a dit to the right of position ℓ in the appropriate term of the ♣-sequence, $R_1(j) \cap S[C(j)] = \emptyset$. On the other hand, a direct computation verifies that $(\mathbb{Z}/d\mathbb{Z}) \bullet_\ell R_2(j) \subset R_2(j)$ and also that $R_2(j) \cap S[C(j)] = R_2(j)$.

Finally, we note that $(\mathbb{Z}/d\mathbb{Z}) \bullet_\ell R_3(j) \subset R_3(j)$. However, the following partition is in general nontrivial:

$$R_3(j) \ = \ \{R_3(j) \cap S[C(j)]\} \sqcup \{R_3(j) \cap (S - S[C(j)])\} \tag{21}$$

Should $C(j)$ admit no control, we are done. If not, let $m < \ell$ be the control qudit, i.e. $S[C(j)] = \{m\}$. Then

$$R_3(j) \cap S[C(j)] \ = \ \left\{ f_1 \cdots f_{\ell-1} k_\ell k_{\ell+1} \ldots k_n \; ; \; \mathbf{f_m} = \mathbf{c_m}, f_1 f_2 \ldots f_{\ell-1} > c_1 c_2 \cdots c_{\ell-1}, k_* \in \{0,1,\ldots,d-1\} \right\} \tag{22}$$

Hence the $\mathbb{Z}/d\mathbb{Z}$ action respects the partition of Equation 21 as well. □

Lemma A.5 *Let $C(j)$, ℓ, and $S_*(j)$ be as above, with $C(j)$ resulting from $c_1 c_2 \ldots c_{\ell-1}\clubsuit\ldots\clubsuit\clubsuit$ of the $\clubsuit$-sequence. Let $z = \{c_1 c_2 \ldots c_{\ell-1} k 00 \ldots 0 \; ; \; k \in \{1,2,\ldots,d-1\} \cap \mathbb{Z}\}$ be the elements zeroed by $\bigwedge[C(j),V(j)]$. Then $R_1(j) \sqcup R_2(j) \sqcup R_3(j) = R_1(j+1) \sqcup R_2(j+1) \sqcup R_3(j+1) \sqcup z$.*

Proof: We break our argument into two cases based on the value of $c_{\ell-1}$.
Case $c_{\ell-1} < d-1$: The $(j+1)^{\text{st}}$ term of the $\clubsuit$-sequence is is given by $c_1 c_2 \ldots (c_{\ell-1}+1)00\ldots0\clubsuit$. Note that for leaves of the tree, the buffering sequence of zeroes is vacuous.

$$\begin{array}{rcl}
R_1(j+1) & = & R_1(j) \sqcup R_2(j) - z \\
R_2(j+1) \sqcup R_3(j+1) & = & R_3(j)
\end{array} \tag{23}$$

Hence $R_1(j) \sqcup R_2(j) \sqcup R_3(j) = R_1(j+1) \sqcup R_2(j+1) \sqcup R_3(j+1) \sqcup z$.
Case $c_p = d-1$: Suppose instead the jth $\clubsuit$-sequence term is $c_1 c_2 \ldots c_{\ell-2}(d-1)\clubsuit\clubsuit\ldots\clubsuit$, so that the $(j+1)^{\text{st}}$ term is $c_1 c_2 \ldots c_{\ell-2}\clubsuit\clubsuit\clubsuit\ldots\clubsuit$. We note that $\{c_0 c_1 \ldots c_{\ell-2}(d-1)0\ldots0\} \in R_2(j) \cap R_2(j+1)$.[2] Then

$$\begin{array}{rcl}
R_1(j) & = & R_1(j+1) \sqcup R_2(j+1) - \{c_0 c_1 \ldots c_{\ell-2}(d-1)0\ldots0\} \\
R_2(j) & = & z \sqcup \{c_0 c_1 \ldots c_{\ell-2}(d-1)0\ldots0\} \\
R_3(j) & = & R_3(j+1)
\end{array} \tag{24}$$

From the first two, $R_1(j) \sqcup R_2(j) = R_1(j+1) \sqcup R_2(j+1) \sqcup z$. Hence $R_1(j) \sqcup R_2(j) \sqcup R_3(j) = R_1(j+1) \sqcup R_2(j+1) \sqcup R_3(j+1) \sqcup z$. □

Proposition A.6 *$S_*(j) = R_1(j) \sqcup R_2(j) \sqcup R_3(j)$ is the set of zero amplitudes (components) of a generic $|\psi_j\rangle$.*

Proof: The proof is by induction. For $j = 1$, we have

$$R_1(1) = \emptyset, \quad R_2(1) = \{00\ldots0*\}, \quad R_3(1) = \{c_1 c_2 \ldots c_{n-1}* \; ; \; \text{some } c_j > 0\} \tag{25}$$

Hence the entire index set $S = S_*(1) = R_1(1) \sqcup R_2(1) \sqcup R_3(1)$.

Hence, we suppose by way of induction that $S_*(j) = R_1(j) \sqcup R_2(j) \sqcup R_3(j)$ and attempt to prove the similar statement for $j+1$. Now $\bigwedge[C(j),V(j)]$ will add new zeroes to the amplitudes (components) with indices z by Lemma A.5. On the other hand, $\bigwedge[C(j),V(j)]$ will not destroy any zero amplitudes existing in $S_*(j)$ due to the induction hypothesis, Lemma A.4, and Proposition A.1. Thus $S_*(j+1) = R_1(j+1) \sqcup R_2(j+1) \sqcup R_3(j+1)$. □

B   Unitary Circuits From State-Synthesis Circuits

In this appendix, we give an alternate construction of optimal order circuits for unitary evolutions from optimal circuits producing states per §5. We present a constructive procedure for building any unitary $U \in \mathbb{C}^{d^n \times d^n}$ from d^n copies of an optimal state-synthesis circuit and other asymptotically negligible subcircuits using an eigen-decomposition of U. If the state-synthesis circuit is any choice that contains $O(d^n)$ gates, then the resulting $O(d^{2n})$ circuit for U is optimal.

---

[2] So in the application, the amplitude (component) of this index is the single amplitude not zeroed by $\bigwedge[C(j),V(j)]$, but it is immediately afterwards zeroed by $\bigwedge[C(j+1),V(j+1)]$.

Let $\{\lambda_j = e^{i\theta_j}\}_{j=0}^{d^n-1}$ be the eigenvalues of U, with $\{|\lambda_j\rangle\}_{j=0}^{d^n-1}$ a corresponding set of orthonormal eigenvectors. We suppose circuits containing $O(d^n)$ two-qudit gates for unitary operators W_j with $W_j|j\rangle = |\lambda_j\rangle$, $0 \le j \le d^n - 1$. Then for a second set of phasing unitaries P_j, we may write

$$
\begin{aligned}
W_j &= |\lambda_j\rangle\langle j| + \Sigma_{k \ne j}|\psi_{j,k}\rangle\langle k| \\
P_j &= I_{d^n-1} + (e^{i\theta_j} - 1)|j\rangle\langle j|
\end{aligned}
\tag{26}
$$

Then by unitarity, $\langle\psi_{j,k}|\lambda_j\rangle = 0$ for all j,k. Now note that $W_j P_j W_j^\dagger = e^{i\theta_j}|\lambda_j\rangle\langle\lambda_j| + \Sigma_{k \ne j}|\psi_{j,k}\rangle\langle\psi_{j,k}| = e^{i\theta_j}|\lambda_j\rangle\langle\lambda_j| + \Sigma_{k \ne j}|\lambda_j\rangle\langle\lambda_j|$. Similarly by induction, the following equality may be verified:

$$
(W_0 P_0 W_0^\dagger)(W_1 P_1 W_1^\dagger)\ldots(W_k P_k W_k^\dagger) = \sum_{j=0}^{k} e^{i\theta_j}|\lambda_j\rangle\langle\lambda_j| + \sum_{j=k+1}^{d^n-1} |\lambda_j\rangle\langle\lambda_j|
\tag{27}
$$

Then considering the eigendecomposition of U and taking $k = d^n - 1$, we have the following factorization:

$$
U = \prod_{j=0}^{d^n-1} W_j P_j W_j^\dagger
\tag{28}
$$

Now note that the techniques of §4 allow for realization of P_j in $O(n)$ two-qudit gates. Thus, since by hypothesis each W_j admits a size $O(d^n)$ circuit, the circuit corresponding to Equation 28 contains $O(d^{2n})$ two-qudit gates and is asymptotically optimal.

As a remark, the circuit synthesis procedure might take $|0\rangle \mapsto |\lambda_j\rangle$ rather than $|j\rangle \mapsto |\lambda_j\rangle$. However an $O(d^n)$ circuit for the latter extracted from an $O(d^n)$ circuit for the former follows from the similarity transform by a local unitary per §5.

C Optimal Asymptotics for Qudit Chains

The optimal asymptotic of $\Theta(d^{2n})$ also holds for more restrictive gate libraries reflecting a choice of architecture. We note this in passing, focusing on the qudit chain architecture.

In the interest of being brief, we do not use formal definitions. Note that the body shows that the library $\text{LU} \sqcup \{\bigwedge_1(\sigma^x \oplus I_{d-2})\}$ is qudit universal, where $\text{LU} = \otimes_1^n SU(d)$ and we intend any instantiation of $\bigwedge_1(\sigma^x \oplus I_{d-2})$ to be allowed. An *architecture* will here refer to a restriction on this gate library. In particular, one supposes that the qudits correspond to the vertices of some graph, which by hypothesis is connected. Then only the instantiations of $\bigwedge_1(\sigma^x \oplus I_{d-2})$ which correspond to edges of the graph are allowed. Since we may construct qudit SWAP between qudits connected by an edge, the restricted library is also qudit-universal. However, the asymptotics of the library gates might be different from the asymptotics of the standard gates. Loosely, instantiations of $\bigwedge_1(\sigma^x \oplus I_{d-2})$ between qudits $O(n)$ vertices apart will now cost $O(n)$ gates rather than one, since $O(n)$ SWAPs between adjacent qudits are also needed.

The notion of a sub-architecture follows by comparing graphs and subgraphs. Thus, if a qudit chain is the architecture of a linear sequence of qudits with consecutive qudits joined by edges, then the qudit chain is a subarchitecture of a finite square, hexagonal, or cubic lattice. If a sub-architecture contains every vertex, then asymptotics of the smaller architecture are at least as good as those of the larger. For the inclusion only admits more possible two-qudit gates.

Thus consider in particular a *qudit chain*. Suppose further the ordering of the dits implicit in earlier notations, e.g. $d_1 d_2 d_3 \ldots d_n$, is now referring to the architecture as well. Thus given the architectural restriction, using SWAPs we see that an instantiation of $\bigwedge_1(\sigma^x \oplus I_{d-2})$ costs $O(|j-k|)$ architecture gates if controlled on qudit j and targeting qudit k, rather than the old count of one gate. A similar comment applies to any two-qudit gate acting between qudits j, k.

By Appendix B, the $\Theta(d^{2n})$ asymptotic follows if we show that the ♣-Householder reduction requires only $O(d^n)$ architecture-local two-qudit gates. Hence let $a(d,n,k)$ denote the number of length k singly-controlled $\bigwedge_1(V)$ specified by the club sequence, where a two-qudit operator acting on qudits j, ℓ, has length $j + \ell + 1$. As an example, the operator of Figure 3 is length four. Now for most k, a length k operation within the $(n+1)^{\text{st}}$ club sequence results from a sequence of $k - 1$ zeroes in some term of the nth sequence in one of two ways:

- A length n term of the form $d_2 d_3 \ldots d_j 00 \ldots 0 \clubsuit \ldots \clubsuit$ is preprended to become $d_1 d_2 \ldots d_j 00 \ldots 0 \clubsuit \ldots \clubsuit$.

- The length n term of the form $00 \ldots 0 \clubsuit \ldots \clubsuit$ is prepended to become $d_1 00 \ldots 0 \clubsuit \ldots \clubsuit$. Here, $d_1 \neq 0$.

Noting this structure, we produce the following recursion relations, which completely determine $a(d,n,k)$:

$$
\begin{cases}
a(d,n+1,k) &= (d-1) + d\, a(d,n,k) \\
a(d,n,0) &= n \\
a(d,n,k) &= 0 \text{ if } k \geq n-1
\end{cases}
\tag{29}
$$

Now $a(d,n,0) = n$ does not factor into the recursive structure of the other $a(d,n,k)$. Rather, evaluating the recursion explicitly for $1 \leq k \leq n-1$, we obtain

$$
a(d,n,k) = (d-1) \sum_{\ell=0}^{n-k-1} d^\ell = d^{n-k} - 1
\tag{30}
$$

We finally use this recursion to obtain our main result.

Indeed, note that since a length k singly-controlled operation may be realized in $O(k)$ local gates, it suffices to prove that $\sum_{\ell=0}^{n-1} k\, a(d,n,k) = \sum_{\ell=0}^{n-1} k(d^{n-k} - 1)$ is a function within $O(d^n)$. This follows by either deriving the appropriate geometric series in order to obtain the exact sum $\sum_{\ell=0}^{n-1} k d^{n-k}$ or alternately by integral comparison of this second sum. Thus, even in the chain gate library, the $\clubsuit$-Householder reduction requires no more than $O(d^n)$ gates and hence recovers an optimal state-synthesis asymptotic of $\Theta(d^n)$. Consequently, Appendix B produces an asymptotic of $\Theta(d^{2n})$ chain architecture-restricted gates for any unitary evolution $U \in U(d^{2n})$.